LOCAL URL SHORTENER WITH WEB INTERFACE

By: Atharv Dubey (NITJ ’28)

**General Information:**

The local URL Shortener works by hosting local servers onto the user system which listen to calls to ‘localhost:5000’ and ‘localhost:8000’ where the former handles the redirection logic and the latter deals with transfer of input and output from front-end to back-end and vice versa. The project is built to learn about transfer of information from one environment to the other and to get an elementary idea about how a server functions. The shortener works on the concept of **Base62 Encoding**. One of the initial aims was to integrate a database system to store information (sqlite), however due to time constraints the project uses a simple text file to store long and short URLs currently, which will eventually be updated to use the intended systems with later version(s). The next update will implement various ‘premium’ features, one such feature is custom URLs for paid subscribers.

**Languages Used:**

* C++ Handles the URL shortening logic
* Python Both the server scripts are built using Python
* HTML/CSS To build and design the web interface
* JavaScript Provides functionality to the webpage form

**External Libraries Used:**

* Python ‘websockets’ and ‘flask’

**Prerequisites:**

* A working python installation ([Download Python | Python.org](https://www.python.org/downloads/))

Note: If any external library is missing from your system, the server scripts will automatically download and install the missing library given an active internet connect and that the python installation is added to ‘PATH’ in system environment variables.

**Usage Instructions:**

* Run ‘server.py’ and ‘server2.py’ scripts. This will load the two servers and a console for each server will be visible.
* Launch ‘index.html’ webpage using any browser.
* Input a URL and press ‘Shorten’ button, this will show a shortened URL of format ‘localhost:5000/(placeholder)’.
* Copy this URL and input it into the search bar of a browser while ‘server2.py’ is running. This will redirect the user to the original URL.